

## Organizacija i polaganje kursa

Ovaj dokument sadrži opis strukture i organizacije kursa, kao i sam način polaganja. Ukoliko imate dodatnih pitanja slobodno se obratite BubbleBee timu.

Kurs se sastoji od skupa lekcija, od kojih su neke uslovljene drugim (formalno, lekcije čine usmeren acikličan graf, što je samo jedna od struktura koje kurs pokriva). Lekcije i njihova uslovljenost će biti prikazane u obliku grafa, tako da lakše možete pratiti progres. Lekcije su podeljene po nivoima, koji definišu generalnu težinu odnosno kompleksnost.

Svaka lekcija sadrži tri sekcije:

- **Materijali** – Ova sekcija sadrži niz video i pisanih materijala koji pokrivaju teoriju, primere i specijalne tehnike vezane za samu temu lekcije. Pre rada na drugim sekcijama preporučujemo da ove materijale detaljno pregledate.
- **Problemi** – Nakon pregledanog teorijskog dela, potrebno je rešiti određeni broj problema. Problemi su birani tako da pokrivaju sve što je potrebno da znate vezano za konkretnu lekciju. Sami problemi su podeljeni u tri dela: *osnovni*, *trening* i *master*.
- **Teorijski test** – U ovoj sekciji ćemo za vas generisati test sa teorijskim pitanjima za konkretnu lekciju. Svo potrebno znanje za rešavanje testa biće pokriveno materijalima iz lekcije.

Da biste završili odredjenu lekciju, potrebno je da uspešno rešite sve probleme iz delova *osnovni* i *trening*, kao i da na teorijskom testu imate više od odredjenog broja poena (zavisno od lekcije). Da biste počeli sa radom na konkretnoj lekciji, potrebno je da pre toga rešite sve lekcije od kojih ona zavisi.

**Problem.** Svaki od problema sadrži detaljni opis samog problema (šta se traži), opis ulaza (koja je tačna struktura ulaza), opis izlaza (koja je tačna struktura izlaza), primer (primer ulaza i izlaza) i ograničenja (ograničenja za ulazne parametre problema koji su navedeni u problemu). Pored ovog problem može sadržati i dodatna objašnjenja primera kao i generalne napomene.

Da bi problem bio uspešno rešen potrebno je napisati program koji će uspešno proći sve test primere u našoj bazi. Naime, za svaki od problema generisan je određen broj test primera. Test primer se sastoji od ulaza i tačnog izlaza (strukture ovih fajlova zadovoljavaju uslove ulaza odnosno izlaza). Vaš kod će biti kompajliran i pokrenut za svaki od ovih test primera. Na standardni ulaz će biti prosleđen ulaz test primera, a vaš izlaz će biti upoređen sa tačnim izlazom test primera. Pored ovoga, vaš program mora da zadovolji i odredjena vremenska i memorijska ograničenja, koja su takođe data u samom problemu.

Primer jednostavnog problema:

**Tekst problema**

Napisati program koji za data dva broja  $A$  i  $B$  vraća njihov zbir.

**Opis ulaza**

Prvi i jedini red standardnog ulaza sadrži dva prirodna broja  $A$  i  $B$ , dovojena jednim znakom razmaka.

**Opis izlaza**

Prvi i jedini red standardnog izlaza treba da sadrži jedan prirodni broj koji je jednak zbiru datih brojeva.

**Primer**

Ulaz	Izlaz
5 13	18

**Ograničenja**

- $1 \leq A, B \leq 10^9$

**Teorijski test.** Teorijski test neće biti obavezan za svaku lekciju. Sadržaće izvestan broj pitanja, koja mogu biti forme pitanja sa ponuđenim odgovorima ili mesto za unos tačnog odgovora. Za uspešno rešavanje potrebno je da u zadatom vremenu sakupite određen broj bodova (zavisno od lekcije). Nakon testa, za svaki netačan odgovor imaćete objašnjenje zašto i gde ste pogrešili. Ukoliko niste zadovoljili date kriterijume, imaćete mogućnost da ponovo polazete test (koji neće biti isti kao prethodni), ali sa pauzom od četiri sata.

Primer teorijskog pitanja:

**Pitanje**

Šta ispisuje rekurzivna funkcija

```
function f (Integer n, Integer m)
begin
    if (n == 0) then
        WriteLine (m);
    endif;
    if (n MOD 2 == 0) then
        f (n - 1, m + 1);
    else
        f (n, m - 1);
    endif;
end;
```

ukoliko je pozvana sa parametrima  $n = 4$  i  $m = 3$ .